A data processing system includes a communication subsystem having an I/O microprocessor for communicating with a central processing unit and a main memory; and a line microprocessor for communicating with a number of devices. The I/O microprocessor and the line microprocessor communicate with each other through mailboxes stored in a shared memory 44. The line microprocessor is controlled by a PROM 58 and channel control programs (CCP's) in a RAM 60, and uses a work RAM 52 for data storage. The line microprocessor interrupts the I/O microprocessor to process data bytes being transferred between main memory and a device requesting service when the line microprocessor has responded to the requesting device and loaded the mailbox.
This invention relates generally to a data processing system in a communications environment, and more specifically to communication multiplexers.

Data processing systems are coupled to a multiplicity of communication lines by a communication controller, commonly called a communication multiplexer. Each of the communication lines is coupled to a device such as a cathode ray tube display (CRT). The communication controller is operative to transfer data between a main memory of the data processing system and the devices via the communication line.

Known communication multiplexers include a first processor for effecting the assembling of data bits from a plurality of communication lines and a second processor for effecting the transfer of data characters between the first and the data processor subsequent to the receipt of one or more special control characters. This had the problem of limiting the throughput and requiring extensive hardware, since the single bit processor was required to assemble the bits into bytes by comparing the bits received from each line with a special stored character.

The hardware required can be reduced by the use of a microprocessor controlling communication lines through the use of channel control blocks stored in a memory. Such a system, however, limited the throughput by restricting the number of communication lines that could be processed.

Accordingly, it is a primary object of the invention to provide an improved communication subsystem for use in a data processing system.
Accordingly, the present invention provides a data processing system for transferring data bytes, comprising a main memory coupled via a system bus to a communication multiplexer coupled to a plurality of input/output device for transferring data bytes between the main memory and the input/output device, characterized in that the communication multiplexer comprises:

shared memory means for storing address and control information and the data bytes;

one microprocessor means responsive to one of the input/output devices requesting service during a polling operation to transfer data bytes between the shared memory means and the input/output device, and having first means for generating a first interrupt signal; and

I/O microprocessor means responsive to the first interrupt signal for transferring data bytes between the shared memory means and the main memory.

A data processing system embodying the invention will now be described, by way of example, with reference to the drawings,

in which:

Figure 1 is a block diagram of the data processing system.

Figure 2 is a block diagram of the communication controller 10.

Figure 3 is a logic diagram of the I/O microprocessor 36 and line microprocessor 56 interrupt logic.

Figure 4 identifies the address locations of the read only memories and the random access memories in communication controller 10.

Figure 5 shows the layout of the mailboxes in shared memory 44.

Figure 6 is a flow diagram showing the I/O microprocessor 36 and line microprocessor 56 in a typical operation.
INTRODUCTORY SUMMARY

The data processing system includes a central processing unit 2, a main memory 4 and a communication subsystem 8, all coupled to a system bus 16. The communication subsystem 8 includes a communication controller 10 and a number of line adapters 12, 14 for coupling a number of devices 18-24 via communication lines to the communication controller.

The communication controller includes a line side and an I/O side. The line side is coupled to a shared memory 44 by a line address bus 70 and a line data bus 72. The I/O side is coupled to the shared memory by an I/O address bus 68 and an I/O data bus 74. The line side controls the transfer of data between the line adapters and shared memory and the I/O side controls the transfer of data between the shared memory and main memory or the central processing unit.

In a typical operation, an area in shared memory called a mailbox may be loaded by the line side with a data byte from one of the devices and the channel number of the device. The I/O side will transfer the data byte stored in the mailbox and a main memory address stored in a channel control block in shared memory to main memory via the system bus. If the device requires a data byte, the line side loads the mailbox with the channel number. The I/O side transfers a main memory request to main memory via the system bus, receives the data byte from main memory and stores the date byte in the mailbox. The line side transfers the data byte from the mailbox to the requesting device.

A line microprocessor 56 controls the operation of the line side and an I/O microprocessor 36 controls the operation of the I/O side. Signals from the line adapters, in response to a poll, cause the line microprocessor to interrupt its operation to service the requesting device, store the information in the mailbox and set a flag bit in the mailbox. The line microprocessor then generates signals to interrupt the I/O microprocessor. The I/O microprocessor transfers the information to main memory, loads a response into the mailbox, and resets the flag bit. The line microprocessor
tests the flag bit and processes the information in the mailbox when the flag bit is reset indicating that the I/O microprocessor has completed the I/O side operation.

DETAILED DESCRIPTION

Figure 1 is a block diagram of the overall data processing system including a central processing unit (CPU) 2, a main memory 4, a communication subsystem 8 and a peripheral controller 6, all coupled to a system bus 16.

The communication subsystem 8 may be operative with a maximum of 16 communication lines and includes a communication controller 10 coupled to system bus 16, and a plurality of line adapters with their associated devices coupled to the communication controller 10 by a line adapter bus 17. The subsystem 8 includes a line adapter 12 having RS232 interfaces and a line adapter 14 having RS422 interfaces; either of these may be replaced by a line adapter 13 having current loop interfaces. Line adapters 13 and 14 may each service up to eight asynchronous lines; line adapter 12 may service up to eight asynchronous lines or up to six asynchronous lines and one synchronous line. However, only two line adapters servicing a maximum of 16 lines may be included in the communication controller 10.

Devices operative with the communication controller 10 include cathode ray tube displays (CRT) 18, a dialing unit (801C) 20, a MODEM (202C) 22, a teletype unit (TTY33) 21, and a line printer 24.

Figure 2 is a block diagram of communication controller 10 which includes an I/O microprocessor 36 which controls the operation of the communication controller 10 with CPU 2 and main memory 4 over system bus 16, and a line microprocessor 56 which controls the operation of the communication controller 10 with the line adapters 12 and 14 over line adapter bus 17.
The I/O microprocessor 36 and the line microprocessor 56 communicate with each other through a shared random access memory (RAM) 44 which stores the line control tables (LCT) and communication control blocks (CCB) and a number of mailboxes. Each of the devices is assigned an LCT. Half of the LCT controls the device in a receive mode and the other half of the LCT controls the device in a transmit mode. Similarly, each of the devices is assigned a CCB for each receive block transfer with main memory 4, and a CCB for each transmit block transfer with main memory 4.

Line control tables include the number of bits in the device data character's length, whether the characters have odd or even parity, the cycle redundancy check (CRC) formula used and the CRC bytes being developed, the status of the device, and pointers to enable the LCT to be operative with a channel control program (CCP).

The CCB stores the main memory 4 address location for the next character either transmitted or received, and the number of characters remaining for processing in the current block. The CCB also stores a control word indicating the last block for transmission, if the CCB was executed, and whether to generate and interrupt upon completion of a block, and a number of status bits indicating the line status at the time the CCB completed. Up to four receive CCB's and up to four transmit CCB's may be stored per device.

A programmable read only memory (PROM) 38 stores the programs that operate with the I/O microprocessor 36. The I/O microprocessor generates signals indicative of an address location in PROM 38 and sends the signals via an I/O paging logic 34 and I/O address bus 18 to PROM 38. An instruction
at that address location is transferred from PROM 38 to the I/O microprocessor 36 via an I/O data bus 74. The I/O microprocessor 36 executes that instruction and generates the address signals indicating the next address location of PROM 38 to read the next instruction over I/O data bus 74.

A work RAM 40 operates with the I/O microprocessor 36 as a scratchpad memory for storing variable data, for stack operation, i.e., storing the return address of an interrupted microprogram, and for providing working storage for data manipulation.

The I/O paging logic 34 receives a virtual address from I/O microprocessor 36 when the I/O microprocessor addresses the LCT or CCB area of shared memory 44 and generates a read address for identifying a location within the LCT or CCB area of a particular channel associated with a selected device.

A bus interface 30 couples the communication controller 10 to the system bus 16 for operation with main memory 4 and CPU 2, and operates with bus request, bus acknowledge, and bus priority operations of known type. Bus interface 30 also provides storage for data and I/O commands that are transferred over system bus 16.

A RAM 60 stores the channel control program (CCP) which processes the data stream of a communication channel. A CCP pointer in the LCT points to the next CCP location in RAM 60 to be referenced by the channel when a channel request interrupt is serviced. The CCP typically controls the transfer of characters between a line adapter interface 66 and shared RAM 44 through line microprocessor 56 and performs the check redundant character computation and minor editing.
A PROM 58 stores the programs that operate with the line microprocessor 56. The line microprocessor 56 generates address signals indicating an address location in PROM 58 and sends the address signals via a line paging logic 54 and a line address bus 70 to PROM 58. An instruction at that address location is transferred from PROM 58 to line microprocessor 56 via a line data bus 72. The line microprocessor 56 executes that instruction and generates the address signals indicating the next address location of PROM 58 to read the next instruction over line data bus 72.

A work RAM 52 operates as a scratchpad memory for line microprocessor 56 as does work RAM 40 for I/O microprocessor 36.

Line paging logic 54 receives a virtual address which is converted into a real address when addressing the LCT or CCB area in shared RAM 44. As with I/O paging logic 34, line paging logic 54 allows a single program to address the LCT or CCB associated with any communication channel (2 channels per line: a receive channel and a transmit channel).

An S register 50 is a one byte index register which is operative with PROM 58.

A pause timer 62 detects if a CCP is running too long by counting the number of accesses to RAM 60. If the number of accesses exceeds a predetermined number, typically 100, the line microprocessor 56 is interrupted, the CCP is temporarily deactivated, and a CCP return address is stored in a queue in work RAM 52.
A priority scan 64 accepts data requests associated with each channel of the device adapters and establishes the priorities for servicing the channels in a suitable sequence.

A line adapter interface 66 couples the line adapters 12 and 14 to the communication controller 10 through line adapter bus 17.

The I/O microprocessor 36 implements a number of functions including handling of I/O instructions from the CPU 2 to the communication controller 10 and controlling the transfer of data between line microprocessor 56 and main memory 4. Line microprocessor 56 in conjunction with PROM 38 acts as an interpreter of the CCP. When a CCP instruction which requests a byte to be transferred to/from main memory 4 is decoded by line microprocessor 56, it stores the number of the channel currently being serviced and the data byte (if the transfer is to main memory) in the mailbox in shared memory 44. The line microprocessor 56 generates an interrupt through an interrupt logic 78 to the I/O microprocessor 36. The I/O microprocessor 36 in conjunction with PROM 38 will address the mailbox in shared RAM 44 for the channel number and command code (and the data byte if this is a receive operation) and address the current CCB of this channel via I/O paging logic 34 for the current main memory address. The I/O microprocessor 36 will transfer the address and data byte to bus interface 30 where the main memory 4 address and data byte are stored, awaiting an acknowledge in response to a bus request, for transfer to main memory 4.

Interrupt logic 78 is also responsive to signals from bus interface 30 to interrupt the I/O microprocessor to accept system
bus 16 information addressed to communication controller 10. The logic 78 is also responsive to a signal from pause timer 62 to interrupt line microprocessor 56 when the number of CCP instructions exceeded the predetermined number; a signal from priority scan 64 to interrupt line microprocessor 56 to start a polling of the devices; and a signal from line adapter 66 to interrupt the line microprocessor 56 when a device responds to the poll.

The I/O microprocessor 36, in conjunction with a free running timer 32, may indicate to the line microprocessor 56 that it is to start a predetermined operation after a time delay determined by the line microprocessor 56.

A clock system 76 generates the phase 1 and phase 2 clock signals for I/O microprocessor 36 and line microprocessor 56 as well as a number of timing signals which are described.

The I/O microprocessor 36 when receiving an I/O command from CPU 2 may generate an I/O instruction to line microprocessor 56 through a mailbox in shared memory 44 in order to control the CCP stored in RAM 60.

A transceiver (XCVR) 46 and a XCVR 48 isolate the I/O data bus 74 from the line data bus 72. Similarly, a MUX and control 42 isolates the I/O address bus 68 from the line address bus 70 and couples shared RAM 44 to either I/O address bus 68 or line address bus 70.

Figure 3 shows the interrupt logic 78 in detail, together with some of the surrounding units. A signal LREADY-O1 or LREADY-O2 at 0 indicates that a device on a communication line coupled to line adapter 12 or 14 requests service by responding
to the poll by priority scan 64. Forcing signal LREADY- to 0 resets a flip-flop 100 on the rise of a clock signal PRICLK-. Output signal LRDYSY- at 0 is applied to an input of a NAND gate 102. Signal STLOAD-, the output of priority scan 64, is at 0 during the polling operation.

A flip-flop 106 is set on the next rise of clock, signal PRICLK- since the D input signal HITVAL+, the output of a NAND gate 102 is 1. This forces the output signal UP2IRQ- to 0, thereby forcing line microprocessor 56 into an interrupt sequence. Line microprocessor 56 generates addresses FFF8 and FFF916 on address lines U2ADO+00 to U2AD15+00 through 16 line paging logic 54 onto line address bus 70 and begins to process instructions stored in PROM 58 and CCP's stored in RAM 60. Signal PRSCCP- is forced to 0 by logic responsive to address signals FFF816 and FFF916 in line paging logic 54. This sets a flip-flop 108. Signal CCPRUN- at 0 resets flip-flop 106 and signals the priority scan 64 that the CCP is active. Interrupt signal UP2IRQ- is forced to 1.

The CCP's control the operation of the communication lines. Each instruction of the CCP calls for a program routine in PROM 62. Line microprocessor 56 performs the instructions of the program routine to perform the CCP instruction.

When the line microprocessor 56 has completed its operation with the communication line, it generates an address OOF116. Line paging logic 54 is responsive to this address and generates signal LNMREF- at 0. A decoder 164 is activated and signal LRQIRQ- is forced to 0. This sets a flip-flop 166 and output UP1IRQ- at 0 forces I/O microprocessor 36 into an interrupt mode.
I/O microprocessor 36 generates addresses FFF8\textsubscript{16} and FFF9\textsubscript{16}. Signal UICRIQ\textsubscript{-} from I/O paging logic 34 is responsive to address FFF8\textsubscript{16} and resets flip-flop 166. The I/O microprocessor 36 is controlled by the program routine stored in PROM 38 to process data in accordance with command signals stored in the mailbox in shared memory 44 by line microprocessor 56.

A CPU 2 may control the communication controller 10 by sending I/O commands over system bus 16. These I/O commands set up the LCT's and CCB's or read the LCT's and CCB's. As an example, one input/output command will set the main memory 4 address in a CCB. Another input/output command will set the range in that CCB.

Bus interface 30 generates signal IOCMMD\textsubscript{+} when an input/output command is received from CPU 2 over system bus 16. Output signal IOCMMD\textsubscript{+} sets a flip-flop 128 on the rise of timing signal MYD100\textsubscript{+} from bus interface 30. Interrupt signal UP1NM\textsubscript{-} at 0 is applied to the non-maskable interrupt input terminal of I/O microprocessor 36 which generates interrupt vector addresses FFFC\textsubscript{16} and FFFD\textsubscript{16}. The input/output command includes a function code which modifies interrupt vector address FFFC\textsubscript{16} in the I/O paging logic 34 to point to an address location in PROM 38 which stores the starting address of the program which executes the input/output command specified by the function code. Interrupt vector address FFFC\textsubscript{16} generates signal NM1CLR\textsubscript{-} in I/O paging logic 34 to rest flip-flop 128.

A flip-flop 126 sets on the rise of the TBORW1\textsubscript{-} signal from pause timer 62 when the pause timer 62 times out. Interrupt signal UP2NMI\textsubscript{-} at 0 is applied to the non-maskable interrupt terminal of line microprocessor 56 which generates interrupt vector addresses FFFC\textsubscript{16} and FFFD\textsubscript{16}. The contents of these
address locations in PROM 58 generate a program address for processing the pause timer 62 time out. Flip-flop 126 is reset during a stop timer or a WAIT instruction when signal PTMRSB-, the output of decoder 164, is forced to 0.

A number of timing and control signals are applied to the input terminals of I/O microprocessor 36 and line microprocessor 56. Signals P1PHZ2+, P2PHZ1+ and P2PHZ2+ applied to the Φ1 and Φ2 terminals provide the basic timing. Signal CKPHZA- applied to the F2 terminal enables the data bus during a microprocessor write cycle and disables the data bus during a microprocessor read cycle. The signals P1HALT- and P2HALT- at 0 applied to the HALT terminal will halt the microprocessor after the instruction is executed. Signal MSTCAD- applied to the R terminal starts the microprocessor when power is turned on.

Figure 4 identifies the address locations of the various memories that are operative with either the I/O microprocessor 36 (I/O side), the line microprocessor 56 (line side), or both the I/O microprocessor 36 and the line microprocessor 56 (shared). Work RAM's 40 and 52 are responsive to address signals 000016 to 03FF16 received from I/O address bus 68 and line address bus 70 respectively.

Shared memory 44 is responsive to address signals 040016 to OFFF16 received from I/O address bus 68 or line address bus 70. Shared memory 44 has 3,072 address locations: 1,024 address locations for storing CCB's for 16 communication lines, 1,024 address locations for storing LCT's for 16 communication lines, 10 address locations for storing mailboxes, and the remaining address locations for storing extra LCT's. Each communication line is operative with CCB's 44a in 64 address locations: 32 address locations for the communication line as a
receive channel and 32 address locations for the communication line as a receive channel and 32 address locations for the communication line as a transmit channel. Each receive channel CCB and each transmit channel CCB includes 8 bytes: 3 bytes of main memory 4 address location, 2 bytes of range, 1 byte of control and 2 bytes of status. Each LCT 44c includes 32 address locations of receive channel configuration and control information and 32 address locations of transmit channel configuration and control information.

RAM 60 includes 16,384 locations with addresses 1000₁₆ to 4FFF₁₆ for storing CCP instructions which are under the control of line microprocessor 56.

PROM 38 has 3,072 locations with addresses F400₁₆ to FFFF₁₆ for storing program instructions that operate with I/O microprocessor 36. PROM 58 has 4,096 locations with addresses F000₁₆ to FFFF₁₆ for storing program instructions that operate with line microprocessor 56.

Each channel has associated with it four 8 byte CCB's 44b, each including 3 bytes of main memory 4 address of the next data byte to be processed by that channel, 2 bytes of range, the number of data bytes remaining in the field, 1 control byte and 2 status bytes.

The CCB control byte includes an "interrupt on status complete" bit, a "valid CCB" bit, and a "last block" bit.

The CCB final status bytes include bits indicating:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The CCP executes a CPU 2 interrupt instruction.</td>
</tr>
<tr>
<td>1</td>
<td>An interrupt was generated for this CCB.</td>
</tr>
</tbody>
</table>
Data Service Error

CCB was executed and status is complete.

CCB Service Error since CCB not available

Flag between CCP and CPU 2.

Data Clock Error

Range not equal to zero when in receive mode. Last block bit of CCB control word set in transmit mode.

Data set status changed.

Memory 4 error corrected.

Invalid memory 4 address.

System bus 16 parity error.

Uncorrected memory 4 error.

The I/O microprocessor 36 and line microprocessor 56 communicate with each other by means of mailboxes stored in locations in shared RAM 44. The contents of these mailbox locations are shown in Figure 5.

The communication controller 10 uses three mailboxes 1 to 3 for a) block mode commands, b) I/O microprocessor 36 commands to line microprocessor 56, and c) line microprocessor 56 commands to I/O microprocessor 36 respectively.
The CPU 2 may initiate a block read operation or a block write operation by means of I/O commands. As a result of the input/output command when the mailbox is available ($F = 0$), the block mode command mailbox is set up with an address in the line microprocessor address space. This is the address of the first location to receive a byte from shared memory 44 if the $D$ bit, bit 7 of word 0, is 0, or from which to transmit a byte to shared memory 44 if the $D$ bit is 1.

Bits 3 to 6 of word 0 specify the channel number of the communication line requiring the block transfer. The CCB stored in shared memory 44 associated with that channel specifies the starting main memory 4 address and the range, the number of bytes in the block, involved in the block transfer.

The $R$ bit, bit 1 of word 0, when at 1 specifies a main memory 4 block read operation and when at 0 a main memory 4 block write operation.

The $F$ bit, bit 0 of word 0, is set to 1 by I/O microprocessor 36 to specify that a command is present, and is rest to 0 by the line microprocessor 56 when the command is completed.

The line microprocessor 56 scans word 0 of block mode command mailbox. If bit 0 of word 0 is 1 then the line microprocessor 56 initiates a firmware routine which identifies the channel number and determines whether this is a read or a write operation. If this is a read operation, then the STORE subroutine is processed. If this is a write operation, then the LOAD subroutine is processed. When the range as stored in the CCB for this channel number reaches ZERO, the line microprocessor 56 resets the $F$ bit, bit 0 of word 0, and terminates this block mode operation.
Mailbox 2, for I/O microprocessor 36 commands to line microprocessor 56, specifies the action the line microprocessor 56 is to take and the reason for the action.

Word 0 specifies the action code. Action code 00\textsubscript{16} specifies a stop I/O command which prevents any further channel activity by halting the CCP program and preventing further data-generated channel request interrupts from the channel specified in word 1. Action code 02\textsubscript{16} initializes the channel by clearing the CCB's and LCT's associated with the channel number specified in word 1. Action code 04\textsubscript{16} starts the CCP execution at an address specified by the LCT words 6 and 7 associated with the channel specified in word 1. This LCT address is specified by CPU 2 initially by an I/O command. Action code 06\textsubscript{16} starts the CCP execution as the result of an interrupt from a communication channel. The CC3 for that channel specifies the starting CCP address location.

Word 2 of mailbox 2 specifies the reason code. Bit 0 indicates a channel request interrupt. Bit 1 identifies a data set scan operation. The data scan routine compares the present status with the old status stored in LCT 14. A difference indicates that a particular channel status has changed. The contents of LCT 8 then determine the action the line microprocessor 56 will take. Bit 2 indicates that the timer 62 which was set by the CCP has timed out. Bit 7 indicates the direction of the line, receive or transmit.

The line microprocessor 56 reads the F bit of word 1. When bit 0 is 1, the line microprocessor 56 reads word 0 and branches to a subroutine specified by the action code. Bit 0 of word 1 is reset to 0 when the action is completed.
Mailbox 3, for line microprocessor 56 commands to I/O microprocessor 36, is active during a request by a line adapter 12 or 14 for service, causing the line microprocessor 56 to start the processing of the CCP instruction specified by the commands stored in the mailbox 3.

Bit 0 of word 0 of mailbox 3 at 1 specifies a load (DMA read) from main memory 4 command at an address specified by the CCB of the channel number stored in word 1 of mailbox 3. The data bytes read from memory are stored in word 2 of mailbox 3. Line microprocessor 56 processes the data bytes in accordance with the CCP as each data byte is stored in the mailbox under control of the I/O microprocessor 36.

Bit 1 of word 0 at 1 specifies a store (DMA write) into main memory 4 at an address specified by the CCB of the channel number stored in word 1. The data bytes are stored in word 2 of the mailbox 3 under control of line microprocessor 56 and transferred to main memory 4 via system bus 16 under control of I/O microprocessor 36. Bit 2 of word 0 at 1 specifies a get next block (GNB) command. This indicates to the I/O microprocessor 36 that the block transfer is complete and to clear the CCB control field. Bit 3 of word 0 at 1 results in the I/O microprocessor 36 interrupting CPU 2 and in conjunction with bit position 2 at 1 (GNB) will result in an I/O command from the CPU 2 loading the CCB for another block transfer. Bit 4 of word 0 at 1 indicates a backspace one character operation. A CRT 18 operator may want to correct a character. Bit 5 of word 0 at 1 indicates that the timer 32 is in an "ON" condition. Bit 6 of word 0 at 1 indicates an initialize operation. Bit 7 of word 0 at 1 indicates the backing up of a line. The CRT 18 operator may want to correct a line.
Bit 0 of word 3 indicates a special 200 ms pause timer 62 operation.

Figure 6 is a flow diagram showing the transfer of a data byte from a communications device, typically a CRT 18, via line adapter bus 17, figure 2, communication controller 10 to memory 4 via system bus 16; and from main memory 4, via system bus 16, communication controller 10, via line adapter bus 17 to CRT 18.

The line adapter 12 or 14 generates signal LREADY-01 or signal LREADY-02, figure 3, which is wire OR'd to generate signal LREADY- (block 200). This sets flip-flop 106 which generates signal UP2IRQ- thereby interrupting line microprocessor 56 (block 202). Line microprocessor 56 generates interrupt vector addresses FFF816 and FFF916. The contents of addresses FFF816 and FFF916 in PROM 58 point to a Channel Request Service Routine stored in PROM 58.

In block 204, the channel number including the D bit, indicating if this is a receive or transmit operation, is stored in a register (not shown) in line paging logic 54 along with the active CCB pointer which selects one of 4 CCB's 44a (figure 4). The channel number and active CCB pointer are used by the line paging logic 54 to convert a virtual address from line microprocessor 56 pointing to a particular LCT or CCB for all 64 communication channels stored in shared memory 44 to a read address pointing to the LCT or CCB for the one requesting channel stored in shared memory 44.

A count of FF16 stored in pause timer 62 by line microprocessor 56 (block 210) starts monitoring the duration of the CCP. In block 212, the starting CCP location in RAM 60 which is stored in an LCT associated with the requesting communication channel is addressed.
We assume first that the channel number indicates a receive operation (block 214); that is, the communication controller 10 will receive a data byte from CRT 18 and load the data byte in the B accumulator of line microprocessor 56 (block 218). A CCP store instruction is called from RAM 60 (block 220) and when the F bit of word 1 of mailbox 3 equals 0 (block 222) command 40₁₆ is stored in word 0 (block 224), the data byte is stored in word 2 (block 226), and the channel number and the F bit set to 1 in stored in word 1 of mailbox 3 (block 228).

In block 230, line microprocessor 56 generates address OOF₁₆ setting flip-flop 166 and thereby generating the I/O microprocessor 36 interrupt signal UP1IRQ-, Figure 3. The I/O microprocessor 36 generates interrupt vector addresses FFF₈₁₆ and FFF₉₁₆ in response to the signal UP1IRQ-. The locations specified by the interrupt vector address store the starting address in PROM 38 of the IRQ Interrupt Handler routine. The line microprocessor 56 calls for a CCP WAIT instruction (block 232), which causes the line microprocessor 56 to suspend operation until the next communication channel interrupt signal LREADY-01 or LREADY-02 is generated.

The I/O microprocessor 36 (block 234) stores the channel number read from word 1 of mailbox 3 into I/O paging logic 34. In block 236, the I/O microprocessor 36 reads the command word in mailbox 3, 40₁₆, indicating a store operation and branches to a DMA WRITE routine in PROM 38.

The bus interface 30 (block 238) is set to give a busy response to any system bus 16 request of communication controller 10. The active CCB pointer is stored in a register (not shown) in I/O paging logic 34 (block 240) and in conjunction with the channel number converts virtual addresses into real addresses.
The I/O microprocessor 36 stores (block 242) the main memory 4 address from CCB 44b, Figure 4, and the data byte from word 2 of mailbox 3 in registers (not shown) in bus interface 30. I/O microprocessor 36 generates address OOF716 (Block 244) to request system bus 16 for the transfer of the main memory 4 address and data byte stored in bus interface 30 to main memory 4 for writing the data byte in the specified address location. The main memory 4 address is incremented and the range decremented (block 246) and written into the CCB of the requesting communication channel. The address indicates the main memory 4 address location into which the next data byte from the requesting communication channel is written. The range, which indicates the number of data bytes remaining to be transferred to main memory 4, is tested for "equal to ZERO" in conventional manner.

If the CRT 18 was requesting a data byte from main memory as indicated by the channel number (D bit), then block 212 would call for a transmit operation in the form of a CCP LOAD instruction (block 216).

The line microprocessor 56 (block 248) tests the flag bit F for 0 (indicating that the I/O microprocessor 36 has completed a previous operation) and sets the load command 8016 into the command word 0 of mailbox 3 (block 250). The channel number and flag bit F set at 1 are stored in word 1 of mailbox 3 (block 252).

The I/O microprocessor 36 is interrupted and branches to the IRQ Interrupt Handler Routine in block 254 as described above; meanwhile the line microprocessor 56 waits for the data byte by testing the flag bit F of word 1 of mailbox 3 for 0 (block 256).
The I/O microprocessor 36 (block 258) stores the active CCB pointer and the channel number in I/O paging logic 34, reads the command word (8016) from word 0 of mailbox 3 (block 260) and branches to a DMA READ routine in PROM 38.

The bus interface 30 is set to give a busy response to a system bus 16 request (block 262). The I/O microprocessor 36 stores the channel number from word 1 of mailbox 3 and the main memory 4 address from the CCB in bus interface 30 registers (not shown) (block 264) and (block 266) requests system bus 16 by generating address OOF716 for the transfer of channel number and address to main memory 4 over system bus 16.

The data byte is received by the I/O microprocessor 36 and stored in word 2 of mailbox 3 (block 268). The flag bit F is set to 0 in work 1 of mailbox 3 (block 270) which indicates to the line microprocessor 56 that the data byte is stored in the mailbox 3. The main memory 4 address is incremented and the range decremented (block 272) for storage in the CCB of the requesting CRT 18.

Line microprocessor 56 tests word 1 for flag bit F equal to 0 (block 274) and (block 276) sends the data byte from word 2 of mailbox 3 out on line adapter bus 17 to CRT 18. In block 278, the line microprocessor 56 returns to the main program.
1. A data processing system for transferring data bytes comprising a main memory (4) coupled via a system bus (16) to a communication multiplexer (10) coupled to a plurality of input/output devices (18-24) for transferring data bytes between the main memory and the input/output devices, characterized in that the communication multiplexer comprises:

shared memory means (44) for storing address and control information and the data bytes;

line microprocessor means (56) responsive to one of the input/output devices requesting service during a polling operation to transfer data bytes between the shared memory means and the input/output device, and having first means for generating a first interrupt signal; and

I/O microprocessor means (36) responsive to the first interrupt signal for transferring data bytes between the shared memory means and the main memory.

2. The system of claim 1 characterized in that the shared memory means comprises:

mailbox means for storing the data bytes and control information;

and channel control block means for storing a plurality of addresses identifying locations in the main memory for storing data bytes received from or being transferred to each of the input/output devices.

3. The system of claim 2 characterized in that the control information comprises: a receive channel number or transmit channel number indicative of the communication multiplexer receiving data bytes from or transmitting data bytes to the input/output device; a load command indicative of the input/output device requesting a data byte from the main memory; a store command indicative of the input/output device transferring data byte to the main memory; and a flag bit in a first state.
indicative of the mailbox means being available to the line microprocessor means, and in a second state indicative of the mailbox means being available to the I/O microprocessor means.

4. The system of claim 3 characterized in that the channel control block means is responsive to the receive channel number to identify the input/output device for selecting a first of the addresses for writing a first data byte received from the input/output device into a first location in the main memory, and responsive to the transmit channel number to identify the input/output device for selecting a second of the addresses for reading a byte from a second location of the main memory for transfer to the input/output device.

5. The system of either of claims 3 and 4 characterized in that the line microprocessor means comprises: a line microprocessor; second means responsive to a ready signal indicating an input/output device requesting service during the polling operation for generating a second interrupt signal for interrupting the line microprocessor, the line microprocessor being responsive to the receive channel number for storing the store command, the data byte, and the receive channel number in the mailbox means when the flag bit is in the first state, and generating the flag bit in the second state; the first means being responsive to selected address signals from the line microprocessor for generating the first interrupt signal.

6. The system of any one of claims 3 to 5 characterized in that the I/O microprocessor means comprises an I/O microprocessor; third means coupled to the first means and responsive to the first interrupt signal for generating a third interrupt signal for interrupting the I/O microprocessor, the I/O microprocessor
being coupled to the mailbox means when the flag bit is in the second state for reading the receive channel number, the store command and the data byte or the transmit channel number and the load command, the channel control block means being coupled to the I/O microprocessor and responsive to the receive channel number for transferring the first of the addresses to the main memory via the system bus, the I/O microprocessor being responsive to the store command for transferring the first of the data bytes to the main memory via the system bus for storage in the first location indicated by the first of the addresses, the channel control block means being responsive to the transmit channel number for transferring the second of the addresses to the second location in the main memory via the system bus, the I/O microprocessor being responsive to the load command for transferring a preassigned channel number to the main memory via the system bus, the main memory transferring the second of the data bytes to the I/O microprocessor for storage in the mailbox means, that I/O microprocessor setting the flag bit to the first state.

7. The system of claim 6 characterized in that the line microprocessor is responsive to the flag bit in the first state for transferring a data byte from the mailbox means to the input/output device.

8. The system of any previous claim characterized in that the first means is a decoder.

9. The system of any previous claim characterized in that the second means comprises a bistable logic element.

10. The system of any previous claim characterized in that the third means comprises a bistable logic element.
INTERPROCESSOR COMMUNICATION MAILBOXES
(Address Locations in Shared Memory 44)

MAILBOX 1 BLOCK MODE COMMANDS

<table>
<thead>
<tr>
<th>BIT POSITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORD 0 1 2 3 4 5 6 7</td>
</tr>
<tr>
<td>0 FR - L L L L D</td>
</tr>
<tr>
<td>1 SHARED MEMORY 44, WORK RAM 52, RAM 60 (MSB)</td>
</tr>
<tr>
<td>2 SHARED MEMORY 44, WORK RAM 52, RAM 60 (LSB)</td>
</tr>
</tbody>
</table>

F=1 COMMAND PRESENT - SET BY I/O MICROPROCESSOR 36
F=0 COMMAND COMPLETED - SET BY LINE MICROPROCESSOR 56
R=1 MAIN MEMORY 4 BLOCK READ
R=0 MAIN MEMORY 4 BLOCK WRITE
LLLL LINE NUMBER (1 OF 16)
D=0 RECEIVE
D=1 TRANSMIT

MAILBOX 2 I/O MICROPROCESSOR 36 COMMANDS TO LINE MICROPROCESSOR 56

<table>
<thead>
<tr>
<th>BIT POSITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORD 0 1 2 3 4 5 6 7</td>
</tr>
<tr>
<td>0 ACTION CODE</td>
</tr>
<tr>
<td>1 F - - L L L L D</td>
</tr>
<tr>
<td>2 REASON CODE</td>
</tr>
</tbody>
</table>

F=1 COMMAND PRESENT - SET BY I/O MICROPROCESSOR 36
F=0 COMMAND COMPLETED - SET BY LINE MICROPROCESSOR 56

ACTION CODE - 0016 STOP IO
0216 CHANNEL INITIALIZE
0416 START I/O FROM CPU 2
0616 START I/O FROM ALL OTHERS

REASON CODE - Bit 0 CHANNEL REQUEST INTERRUPT
1 DATA SET SCAN
2 TIMER
7 TRANSMIT/RECEIVE (LOGICAL ONE = TRANSMIT CHANNEL/LOGICAL ZERO = RECEIVE CHANNEL)

Fig. 5. (Sheet 1 of 2)
MAILBOX 3 LINE MICROPROCESSOR 56 COMMANDS TO I/O MICROPROCESSOR 36

<table>
<thead>
<tr>
<th>WORD</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>F</td>
<td>-</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>D</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

F = 1 COMMAND PRESENT - SET BY LINE MICROPROCESSOR 56
F = 0 COMMAND COMPLETED BY I/O MICROPROCESSOR 36

COMMANDS - WORD 0

<table>
<thead>
<tr>
<th>BIT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LOAD</td>
</tr>
<tr>
<td>1</td>
<td>STORE</td>
</tr>
<tr>
<td>2</td>
<td>GET NEXT BLOCK</td>
</tr>
<tr>
<td>3</td>
<td>INTERRUPT CPU 2</td>
</tr>
<tr>
<td>4</td>
<td>BACK SPACE ONE CHARACTER</td>
</tr>
<tr>
<td>5</td>
<td>TIMER 32</td>
</tr>
<tr>
<td>6</td>
<td>INITIALIZE</td>
</tr>
<tr>
<td>7</td>
<td>BACK ALINE</td>
</tr>
</tbody>
</table>

WORD 3

<table>
<thead>
<tr>
<th>BIT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>SPECIAL TIMER</td>
</tr>
</tbody>
</table>

\[ \text{Fig. 5. (sheet 2 of 2)} \]
Fig. 6.
(sheet 1 of 2)
**Fig. 6. (sheet 2 of 2)**